

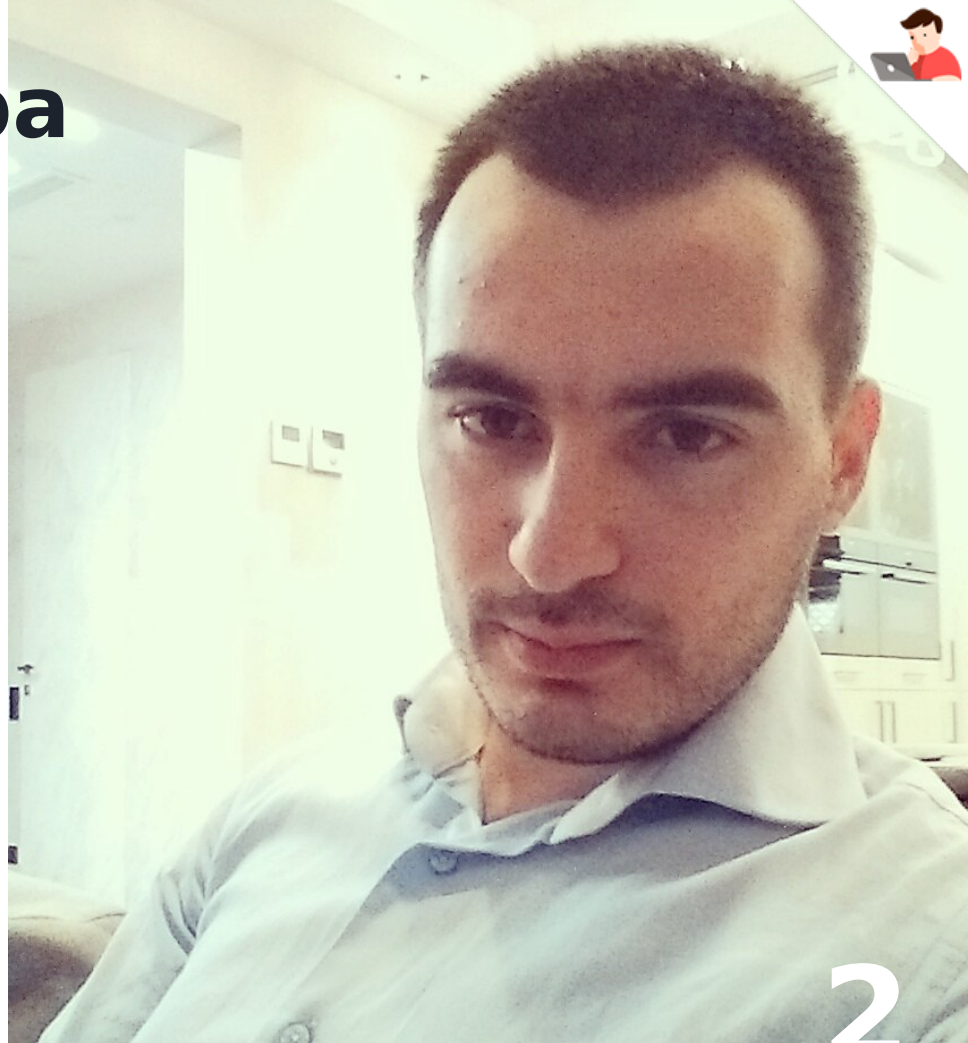
Функции

Ведущий вебинара

Сергей Мелюков

Круг интересов: Backend,
Frontend, GameDev, MobileDev

Место работы: Frontend
разработчик профессиональных
инструментов Avito



Содержание



1. Введение в функции
2. Типы объявления функций
3. Привязка аргументов
4. Замыкания и каррирование
5. Анонимные функции
6. Всплытие переменных и функций

The background features a stylized illustration of mountains in shades of gray and white, with a white cloud on the left. The word "Функции" is centered in a bold, dark gray font. There are yellow L-shaped corner brackets in the top right and bottom left corners, and a yellow horizontal bar at the bottom with a large yellow number "4" on the right side.

Функции

Функции

Функции - это кусочки JS-кода, которые можно *вызывать* повторно.

У функции может быть *имя*, *параметры* и *аргументы*.

Имя функции - имя, по которому мы будем обращаться к этой функции в дальнейшем.

Параметры - *именованные* данные, которые могут быть *переданы* в функцию при ее вызове и *использоваться* внутри функции.

Аргументы - непосредственно данные, которые передаются в функцию при ее вызове.





Параметры функции

К значениям параметров (*аргументам*) можно обращаться как по имени, так и через специальный объект *arguments*.

arguments - служебная переменная, которая доступна внутри любой функции.

- Содержит в себе *значения* переданных в функцию аргументов и пытается вести себя *как массив*.
- Содержит свойство *length*, в котором содержится *количество* переданных в функцию аргументов.

К аргументу можно *обратиться*, указав его *номер* в квадратных скобках.

The background features a stylized illustration of mountains in shades of gray and white. A white cloud is positioned on the left side, partially overlapping the mountain peaks. In the top right corner, there is a yellow L-shaped decorative line. In the bottom left corner, there is another yellow L-shaped decorative line. The title text is centered horizontally and partially overlaid by the cloud.

Типы объявления функций



Типы объявления функций

Declaration - объявление функции в основном потоке кода.

При таком объявлении, интерпретатор *сам* создает переменную и помещает в нее код функции.

Expression - объявление функции внутри *выражения*.

В таком случае, необходимо *самостоятельно* создать переменную и записать в нее код функции.



Возврат значений

Любая функция может **возвращать** результат своей работы при помощи ключевого слова **return**

Результатом может быть всё, что угодно (это решает разработчик).

Используя возврат значений из функции, мы можем более *гибко управлять* результатом работы функции.

The background features a stylized illustration of mountains in shades of gray and white, with a white cloud on the left. Yellow L-shaped corner brackets are positioned in the top right and bottom left corners.

Привязка аргументов



Привязка аргументов до вызова функции

В JS есть возможность **привязать** к функции любое количество аргументов до ее непосредственного **вызова**.

Это значит, что в момент вызова такой функции, привязанные к ней аргументы будут подставлены **автоматически**.

Для этого, у каждой функции в JS есть метод **bind**, который возвращает копию нашей функции, но с указанными привязанными аргументами.

Первым параметром необходимо передать **контекст** (более подробно [здесь](#))

Далее, через запятую, перечисляются **аргументы**, которые мы хотим привязать к функции.



Замыкания и каррирование



Замыкания

Замыкания - способность функции **запоминать** переменные, которые были доступны в той *области видимости*, в которой эта функция была *объявлена*.

При обращении к какой-либо переменной, сначала переменная ищется **внутри функции**, затем в области видимости, где функция была *объявлена*.

Частый пример использования замыканий - **каррирование**:

Если из функции *А* вернуть функцию *Б*, то функция *Б* запомнит аргументы переданные в функцию *А* и сможет их использовать.

The background features a stylized illustration of mountains in shades of gray and white, with a single white cloud on the left side. The overall aesthetic is clean and modern.

Анонимные функции



Анонимные функции

Анонимные функции - функции без имени.

Используются обычно в тех случаях, когда имя функции не важно:

- возврат функции из функции
- передача небольшой функции в качестве параметра

Очень часто, анонимные функции вызывают сразу же после их создания:

```
(function(a, b){return a+b;})(1, 2);
```

Такой способ вызова функций называется: **Immediately Invoked Function Expression (IIFE)**

The background features a stylized illustration of mountains in shades of gray and white, with a white cloud on the left. The overall aesthetic is clean and modern.

Всплытие

Всплытие

Всплытие (hoisting) - побочный эффект **замыканий**.

Мы можем использовать переменные из **родительской** области видимости вне зависимости от того - выше или ниже объявления функции эти переменные были объявлены.

Функции объявленные при помощи **Declaration** могут быть вызваны до того, как будут *объявлены* (в пределах *одной* области видимости).

Функции объявленные при помощи **Expression** ведут себя как переменные и *не* могут быть вызваны до своего объявления



The background is a solid yellow color. It is decorated with a repeating pattern of small, semi-transparent yellow question marks. In the top right and bottom left corners, there are white L-shaped brackets, each consisting of a horizontal line and a vertical line meeting at a right angle.

Время ваших вопросов